



UNIVERSIDAD DE BUENOS AIRES
FACULTAD DE CIENCIAS EXACTAS Y NATURALES
DEPARTAMENTO DE COMPUTACIÓN

Título: tesis

Tesis presentada para optar al título de
Licenciado en Ciencias de la Computación

Guillermo Alejandro Gallardo Diez

Director: Demian Wassermann

Codirector:

Buenos Aires, 2011

Índice general

1.. Introducción	1
2.. Introducción a la Resonancia Magnética de Difusión	5
2.1. Imagen por resonancia magnética	5
2.2. Resonancia Magnética de Difusión	6
3.. Parcelando la Corteza Cerebral	11
3.1. Materiales utilizados en el estudio	11
3.2. Seleccionando voxels que serán semillas	11
3.3. Creando tractogramas a partir de las semillas	12
3.4. Clustering de Semillas: Estado del Arte.	13
4.. Contribuciones	15
4.1. Estudio de la Convergencia de los Tractogramas	15
4.2. Análisis del Método Moreno-Dominguez	15
4.2.1. Clustering de vectores colineales	16
4.2.2. Relación métrica- <i>linkage</i>	16
4.2.3. Complejidad algorítmica del Clustering	18
4.3. Clustering en el Espacio Euclídeo	18
4.3.1. Transformación Logit	18
4.3.2. Modificando el Algoritmo de Clustering	19
4.3.3. Mejorando la Complejidad Espacial: Matrices Ralas	20
5.. Resultados	23
5.1. Estabilidad Tractogramas	23
5.2. Posicionando semillas	24
5.3. Parcelando el Área de Broca	25
5.3.1. Distancia coseno con centroide	25
5.3.2. Utilizando LogOdds	25
5.3.3. Lado a lado	25
5.4. Parcelando la Corteza	25
5.4.1. Utilizando LogOdds	25

1. INTRODUCCIÓN

En la neurociencia moderna existe la teoría de que el cerebro puede ser dividido en áreas de acuerdo a distintos criterios estructurales, siendo la citoarquitectura de Broadmann [1] una de las más conocidas. Actualmente existe evidencia de que es posible atribuir un criterio funcional a cada una de estas regiones, como mover la mano o procesar el lenguaje [2]. No obstante, el cómo se relacionan dichas áreas con funciones complejas sigue siendo un problema abierto [3]. Así mismo, tampoco se conoce si esta división es única, y en tal caso, qué límites definen a cada área. La relación función-estructura posee diversas aplicaciones en la medicina, como por ejemplo el planeamiento quirúrgico [4] [5]; asistencia durante la cirugía [6] y rehabilitación [7] de pacientes. Por ello existe la necesidad de entender cómo es que el cerebro está conectado estructuralmente, esto es, entender qué regiones están conectadas físicamente por axones y cuánto influye esto en el aspecto funcional del cerebro.

El cerebro puede dividirse en varios tipos de tejido, nosotros nos enfocaremos en dos de ellos: materia blanca y materia gris. La corteza está formada por materia gris, la cual es densa en neuronas. Estas neuronas están conectadas entre sí mediante axones. Se denomina materia blanca al tejido donde la proporción de axones es superior a la de neuronas [8]. La reciente invención de la Resonancia Magnética de Difusión (dMRI) permitió desarrollar nuevas técnicas para estudiar la conectividad estructural [9]. En particular, el conocer la intensidad de difusión que existe en cada punto del cerebro permite caracterizar los axones dentro de la materia blanca [10]. Una forma de hacerlo es utilizando algoritmos de tractografía [11] [12], estos toman un punto del cerebro como semilla y devuelven un tractograma. Un tractograma es una imagen donde cada voxel representa la probabilidad de que ese punto del cerebro esté conectado a la semilla elegida mediante un conjunto de axones. Las probabilidades se estiman mediante un procedimiento Monte Carlo, simulando el recorrido de un número de partículas de agua por la materia blanca comenzando desde dicha semilla. Distintos grupos han propuesto el agrupar estas semillas empleando un algoritmo de *clustering* para definir nuevos criterios de parcelamiento de la corteza cerebral.

Los algoritmos de *clustering* son una herramienta de análisis en campos como *Machine Learning* y *Data Mining*. Permiten agrupar objetos en base a algún criterio de similitud. Ejemplos de ellos son:

- K-means: Divide n vectores en k clusters diferentes, siendo k un número predefinido. Cada cluster está formado por los elementos que más cerca están al vector medio del mismo. [13]
- Agglomerative Hierarchical Clustering: Cada observación comienza en un cluster distinto. Luego, el algoritmo selecciona iterativamente dos clusters siguiendo algún criterio de similitud, los agrupa en un nuevo cluster y crea un elemento representativo de este. La jerarquía resultante de agrupar todos los clusters es expresada como un dendrograma. [14]

- Gaussian Mixture: Asume que todas las observaciones provienen de un número finito de distribuciones Gaussianas con parámetros desconocidos. Implementa *expectation-maximization* para determinar dichos parámetros iterativamente. [14]

Cada algoritmo utiliza distintos modelos de cluster, por lo que el espacio de aplicación y el resultado varía significativamente de uno a otro.

Como ya fue dicho, una forma de parcelar la corteza cerebral es haciendo un *clustering* de semillas. El primer paso para poder hacer esto es seleccionar la posición de las mismas. Dado que la materia blanca está compuesta principalmente por axones, es en ella donde se sitúan las semillas. Seleccionar los voxels que serán semilla requiere contar con alguna manera de discriminar entre materia blanca y materia gris en la imagen. A su vez, si queremos que las semillas estén a cierta distancia de la corteza, se debe tener cuidado en respetar la forma del cerebro. En la literatura actual se utilizan hasta veinte mil semillas por hemisferio [15] y cien mil partículas por semilla [16] para generar los tractogramas. Realizar cada tractograma en paralelo reduce el tiempo total del procesamiento del cerebro. Al momento de representarlos es importante seleccionar la estructura correcta. Una matriz del tamaño de la imagen de dMRI es la implementación más sencilla, pero como explicaremos en la sección X, solo el 30 % de los valores serán no nulos. Luego, para agrupar los tractogramas se debe seleccionar un modelo de datos para representar los datos; un algoritmo de *clustering* y una medida de similaridad. Finalmente, una vez obtenidos los clusters, es necesario mapear cada semilla con su respectivo voxel en la corteza cerebral. En resumen, primero hay que situar las semillas en la materia blanca; luego se deben generar los tractogramas de estas y finalmente hay que aplicar el algoritmo de *clustering*. Todos estos pasos son caros en términos computacionales.

Dada la cantidad de opciones a tener en cuenta, distintos grupos han aplicado diferentes criterios de *clustering* para parcelar la corteza. Por ejemplo, Behrens et. al [?] utilizan *Target-Based Clustering*, este algoritmo define como restricción que cada área solo puede estar conectada con alguna otra perteneciente a un conjunto predefinido. Anwander et. al [16] parcelan el Área de Broca utilizando *k-means*, para lo cual se necesita definir un número de parcelas a encontrar. Moreno-Dominguez et al. [15] sitúan semillas en la interfaz entre materia blanca y materia gris, a partir de las cuales se crean tractogramas. Dichos tractogramas son luego agrupados utilizando el algoritmo *Agglomerative Hierarchical Clustering* con la distancia coseno como medida de similitud, y un centroide como representante de cada Cluster. La gran ventaja de este último caso es que la parcelación resultante no posee restricciones respecto al número de parcelas a generar, o sobre como las mismas deberían estar conectadas. Por el contrario, dependiendo la forma en que se desee cortar el dendrograma se obtendrán distintos grados de granularidad en la parcelación. Sin embargo, no queda claro que el criterio utilizado para medir distancias entre clusters y la forma de representar la unión de los mismos sean compatibles. Entonces, si bien el método de Moreno-Dominguez no asume cuantas o como son las áreas a encontrar, aún no posee el suficiente formalismo. Por esto es importante seguir investigando nuevos modelos.

El objetivo de este trabajo es analizar los métodos de *clustering* jerárquico actuales para parcelar toda la corteza y proponer un nuevo enfoque. Utilizamos la base de datos provista por *Human Connectome Project* [17], ésta posee la dMRI de varios sujetos organizada por sexo y edad, con la ventaja de que todos los datos están ya preprocesados.

Esto nos permite enfocarnos en el problema del *clustering*, a la vez que permite reproducir con mayor facilidad nuestro estudio.

Una contribución menor de nuestro trabajo es el análisis de dos implementaciones existentes de algoritmos de tractografía. Dada la naturaleza estocástica de los mismos es importante comprobar si el resultado se estabiliza al utilizar un número suficientemente grande de semillas. También es importante determinar si algoritmos diferentes convergen a una misma solución. En particular mostramos que para dos implementaciones distintas, ambas basadas en la librería *dipy*, los tractogramas convergen. Mas aún, lo hacen a resultados similares.

La contribución principal de este estudio es el realizar un análisis teórico del estado del arte; mostrar las problemáticas que presenta y proponer una nueva forma de parcelar la corteza mediante el *clustering* de semillas. Nuestro método consiste en transformar los tractogramas a un espacio vectorial y luego agruparlos con *Agglomerative Hierarchical Clustering*. Utilizamos la métrica euclidiana como medida de similitud y el centroide como *linkage*. Presentamos una implementación eficiente de nuestro método. También el como adaptarlo al uso de matrices ralas para optimizar el costo espacial. Nuestro método produce parcelaciones similares a las obtenidas usando el estado del arte. La ventaja de nuestro algoritmo es que reduce significativamente el orden de complejidad temporal y espacial.

2. INTRODUCCIÓN A LA RESONANCIA MAGNÉTICA DE DIFUSIÓN

Este capítulo está basado en el libro *Diffusion MRI* [18] y en las clases del Doctor Michael L. Lipton [19] disponibles online. En caso de querer profundizar en algún tema, por favor referirse a estos.

2.1. Imagen por resonancia magnética

Se denomina momento magnético nuclear al momento magnético que posee un átomo a causa del spin de sus protones y electrones. Cuando un protón con momento magnético $\vec{\mu}$ es puesto dentro de un campo magnético, comenzara a precer en torno a la dirección de este último con una frecuencia:

$$\omega = \vec{\mu} \times \vec{B} = \gamma \vec{J} \times \vec{B}$$

Denominada frecuencia de Larmor, donde ω es la velocidad angular; γ es la relación giromagnética del protón; \vec{J} es su momento angular y B es la fuerza del campo. A su vez, la cantidad de energía del campo determinara el ángulo entre μ y el campo mientras sucede la precesión (Figura 2.1). Esto quiere decir que dado un B suficientemente grande es posible hacer que la precesión suceda en la dirección transversal del campo, lo cual permitiría medir $|\mu|$ simplemente poniendo una bobina en ese plano (Figura 2.2). Si uno realizará el experimento notaría que al apagar el campo, la señal comienza a desvanecerse, esto es porque el sistema comienza a perder energía provocando que el ángulo entre μ y el campo se achique. A este proceso se lo denomina relajación.

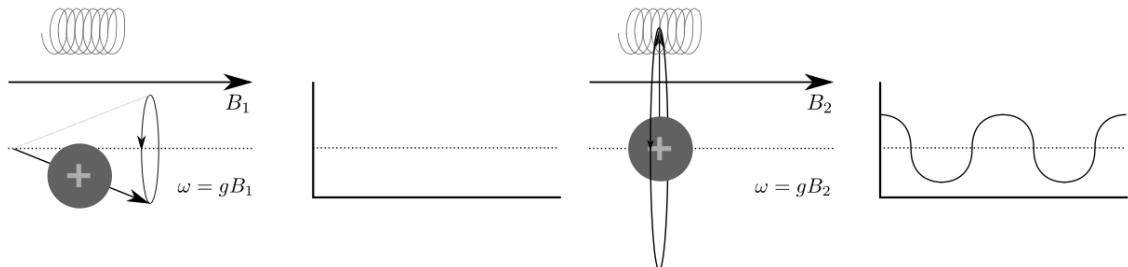


Fig. 2.1: Spin sometido a un campo magnético débil Fig. 2.2: Resultado de aumentar el campo magnético

Vayamos ahora al plano médico. El cuerpo está compuesto por distintos tipos de tejidos, cada uno con su propia composición química, lo cual determina un momento magnético distinto para cada uno de ellos, y por ende un tiempo de relajación particular. Esto implica que al poner a un paciente dentro de un campo magnético, cada tejido comenzara a generar un momento en base a la población de protones que posea. Como ya fue dicho, una forma de medir el tiempo de relajación de estos es utilizando un campo magnético. El problema es que si uno intentara administrar energía al sistema simplemente aumentando la fuerza del campo magnético terminaría dañando al paciente. Aquí es donde se aprovecha la

frecuencia de Larmor. Conociendo la composición química de cada tejido es posible calcular su frecuencia angular. Luego, mediante el efecto de resonancia es posible transmitir energía al sistema simplemente emitiendo ondas en esa misma frecuencia.

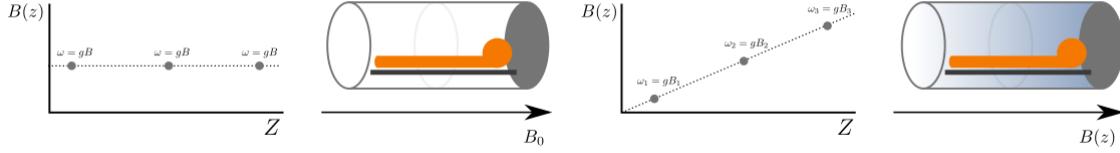


Fig. 2.3: Campo uniforme, todos los protones poseen la misma velocidad angular

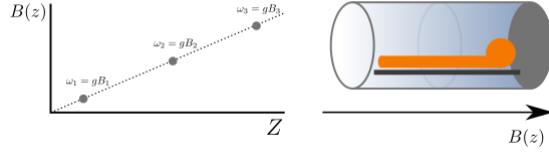


Fig. 2.4: Campo gradiente, la velocidad angular de los protones varía linealmente

Los resonadores magnéticos son dispositivos con la capacidad de generar campos y pulsos en diferentes frecuencias, en particular todo resonador encendido está emitiendo constantemente un campo homogéneo B_0 (ver Figura 2.3). El problema entonces es: cómo obtener el tiempo de relajación de un punto particular del cuerpo. Para esto es posible utilizar camposgradientes. Un campo gradiente es un campo que varía su potencia linealmente a lo largo de una dirección, provocando que todas los protones a lo largo de esa dirección varíen su frecuencia angular de manera predecible. Aplicando un campo gradiente G_z en la dirección z (ver Figura 2.4) sobre el paciente haremos que la velocidad en función de la posición sea: $\omega(z) = B_z(z)g$, esto nos asegurara que si aplicamos el pulso de RF a una frecuencia de $B_z(z_o)g$, solo los protones que se encuentran en la posición $z = z_o$ comenzaran a resonar, por lo que estos serán los únicos que generen un campo transversal. Cabe destacar que como no es posible generar un pulso con exactamente la frecuencia deseada, también resonaran los protones que se encuentren cerca, por lo que tendremos un intervalo $[z_o - \epsilon, z_o + \epsilon]$ resonando. A este proceso se lo denomina *slice selection*. Podemos pensar el slice como una matriz de dos dimensiones sobre el eje z . Si ahora aplicamos un campo gradiente G_ψ en la dirección y , sucederá que todas las filas de nuestra matriz adquiriran diferentes velocidades angulares. Al apagar G_ψ todos los protones volveran a preceder respecto al campo B_0 , pero esta vez estaran desfasados por filas (ver Fig. 2.5). Encendiendo un tercer campo gradiente G_v sobre la dirección x conseguiremos que cada columna posea una frecuencia distinta y cada fila una fase distinta. Repitiendo este procedimiento varias veces cambiando solo la intensidad de G_ψ podemos armar lo que se conoce como *k-space*. El *k-space* es una imagen espacial temporal donde están anotados los valores obtenidos para cada potencia utilizada, en orden ascendente de potencia. El aplicar una transformada de Fourier 3D a dicho espacio nos devolverá la imagen que representa el contraste de cada tejido en el *slice* seleccionado.

2.2. Resonancia Magnética de Difusión

Las moléculas dentro de un fluido en equilibrio no se encuentran estáticas, sino que se mueven realizando un camino aleatorio. A este fenómeno se lo conoce con el nombre de

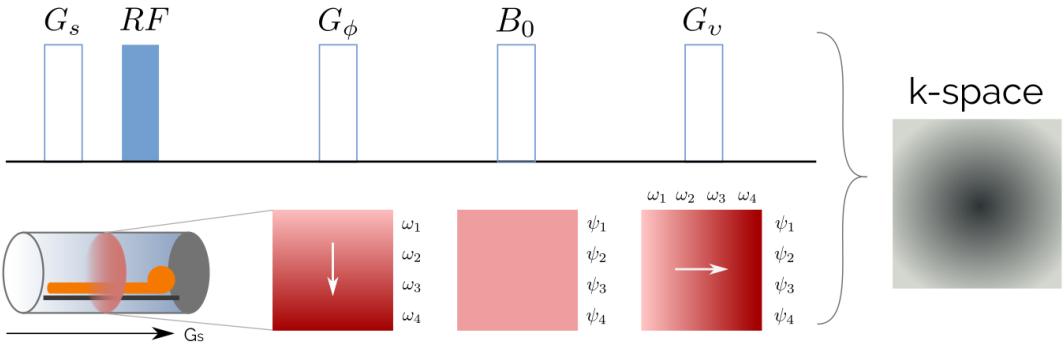


Fig. 2.5: Resumen del proceso de adquisición de imágenes en MRI

difusión.

En 1946 Bloch [20] prueba que la variación de la magnetización nuclear en el tiempo se puede expresar como:

$$\frac{dM(t)}{dt} = \gamma M(t) \times B(t) - \frac{M(t)\vec{x} + M(t)\vec{y}}{T_2} - \frac{(M(t) - M(0))\vec{z}}{T_1}$$

Donde γ es la relación giromagnética, B es la intensidad del campo magnético y T_1, T_2 son tiempos de relajación. Mas tarde, en 1956, H.C. Torrey [21] observa que la magnetización también se pierde por efecto de la difusión y extiende la ecuación de Bloch:

$$\frac{dM(t)}{dt} = gM(t) \times B(t) - \frac{M(t)\vec{x} + M(t)\vec{y}}{T_2} - \frac{(M(t) - M(0))\vec{z}}{T_1} + \nabla \cdot D \nabla M(t)$$

Donde D es el tensor de difusión. Esta relación se conoce como la ecuación de Bloch-Torrey.

Imaginemos el siguiente experimento: luego de aplicar el pulso RF agregamos un campo gradiente $G_1 = G_d$ durante un tiempo δ pequeño. Como ya explicamos, esto generará un desfase entre los spines de los protones. El aplicar $G_2 = -G_d$ luego de Δ debería provocar que los spines se vuelvan a alinear. Sin embargo, los protones que se encuentren en un fluido estarán sometidos al efecto de la difusión. Esto sucede, por ejemplo, en el interior de los axones. Dependiendo del tiempo δ los protones se habrán movido cierta distancia, provocando que el campo magnético los alcance en distintas posiciones. Por ende su velocidad angular se verá afectada de manera distinta a la esperada si no se hubieran movido. Esto nos indica que si hay difusión entonces habrá un desfase en esa población de neutrones (ver Figura 5.20).

Recordemos que la señal medida en un resonador magnético proviene del momento magnético de los protones. Es importante entender que por limitaciones físicas de los dispositivos es imposible obtener la señal producida por un solo protón. Lo que se mide es la resultante de los momentos magnéticos de todos los protones dentro de un espacio. Si todos los protones están precediendo a la misma velocidad sobre el mismo plano, entonces la resultante máxima se obtiene cuando todos poseen la misma fase. Esto es, todos se

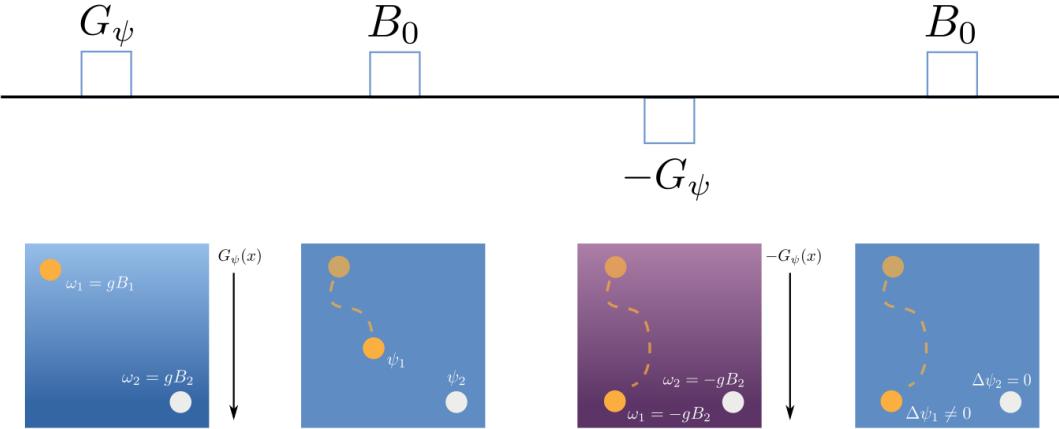


Fig. 2.6: Modificar la secuencia permite medir la intensidad de difusión

encuentran en la misma posición al mismo tiempo, rotando juntos. Por ende, el desfase producto de la difusión se traducirá en perdida de señal.

En 1965 Stejskal y Tanner [22] crean una secuencia denominada *Pulsed Gradient Spin Echo*. En la misma utilizan dos pulsos de RF y un gradiente magnético para generar un desfase entre los protones (ver Figura 2.7). Luego demuestran que asumiendo un δ pequeño la ecuación de Bloch-Torrey tiene solución, y la atenuación de la señal se puede expresar como:

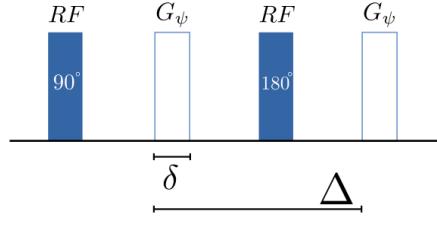


Fig. 2.7: Pulsed Gradient Spin Echo

$$E(g, \delta, \Delta) = \frac{S(g, \delta, \Delta)}{S_0} = e^{-\gamma^2 g^2 \delta^2 (\Delta - \frac{\delta}{3}) D} \quad (2.1)$$

Donde $E(g, \delta, \Delta)$ es la atenuación de la señal obtenida; g es la intensidad del gradiente magnético; S_0 es la señal obtenida sin utilizar gradientes ($g = 0T/m$); γ es la relación giromagnética del protón; δ es el tiempo que el gradiente está encendido; Δ es el tiempo entre las activaciones del gradiente y D representa el coeficiente de difusión. La razón por la cual se divide la señal obtenida por S_0 es porque la señal en cada punto depende fuertemente de la densidad de protones que hay en el mismo, si no ponderamos dicha densidad, es imposible comparar la intensidad de difusión en distintas regiones.

En 1985 Le Bihan [23] adapta esta técnica para medir la difusión de las partículas agrupando todos los parámetros del experimento dentro de un mismo parámetro b :

$$b = \gamma^2 g^2 \delta^2 \left(\Delta - \frac{\delta}{3} \right)$$

Esto simplifica la ecuación 2.1:

$$E(b) = \frac{S(b)}{S_0} = e^{-bD}$$

Donde b representa el reciproco de la intensidad de difusión.

En 1994 Basser et al. [24] proponen medir la atenuación de señal en distintas direcciones y luego aproximar el coeficiente de difusión con un tensor de segundo orden. Un tensor es una matriz multidimensional asociado a una base, que posee una ley de transformación para indicar cómo cambian los componentes del tensor al cambiar de base. Esta técnica sienta las bases de lo que se conoce como *Diffusion Tensor Imaging* (DTI). En DTI el tensor mas utilizado representa un elipsoide en R^3 . La matriz que lo representa es simétrica, por ello es que se necesitan tomar al menos seis adquisiciones:

$$D = \begin{pmatrix} D_{xx} & D_{xy} & D_{xz} \\ D_{xy} & D_{yy} & D_{yz} \\ D_{xz} & D_{yz} & D_{zz} \end{pmatrix}$$

El problema con utilizar este es que no permite representar correctamente el cruce de fibras dentro de cada voxel dado que las caracteriza con un solo elipsoide.

En 1991 Callaghan et al [25] desarrollan el *q-space analysis*. Esto permite realizar microscopía mediante dMRI. Utilizando el trabajo de Stejskal y Tanner prueban que es posible obtener la siguiente relación entre la señal atenuada y una transformada de Fourier:

$$E(q, \Delta) = \frac{S(q, \Delta)}{S_0} = \int_{R^2} p(r; \Delta) e^{-2\pi i q r} dr$$

$$q = \frac{\gamma \delta g}{2\pi}$$

Donde $p(r; t)$ es la densidad de probabilidad de que una población de partículas se desplace en dirección r durante un tiempo t . $p(r; t)$ es característico del compartimiento donde se mueven las partículas.

Una de las principales ventajas de *q-space* sobre DTI es que no asume ningún modelo a priori, esto permite definir distintos tipos de modelos para $p(r, t)$ que caracterizan mejor el cruce de fibras. *Spherical Harmonics* [26] o *Constrained Spherical Deconvolution* [27]. son ejemplos de ello.

3. PARCELANDO LA CORTEZA CEREBRAL

En el capítulo anterior presentamos los fundamentos físicos de la Resonancia Magnética Nuclear y como es posible utilizar estos para caracterizar la difusión en el cerebro. Nuestro objetivo ahora es parcelar la corteza cerebral haciendo uso de un criterio estructural. En particular, utilizando una imagen de difusión, queremos generar una parcelación mediante el agrupamiento de tractogramas. Para ello es necesario primero seleccionar los voxels que serán utilizados como semilla de cada tractograma; luego generarlos y finalmente agruparlos usando algún algoritmo de clustering. Dejamos el siguiente diagrama para ser utilizado como referencia de los pasos a seguir:

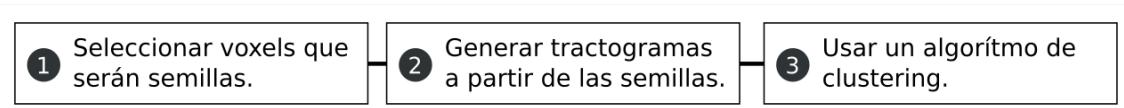


Fig. 3.1: Pipeline del proceso de parcelación

3.1. Materiales utilizados en el estudio

Todos nuestros experimentos fueron realizados con sujetos descargados de la base de datos de *Human Connectome Project*. Las ventajas de utilizar estos datos son muchas: Tanto la imagen de difusión como la anatómica se encuentran ya preprocesadas [28]; cada sujeto posee una parcelación que, entre otras cosas, permite separar la materia blanca de la gris; y cada sujeto posee una superficie que representa su corteza cerebral.

3.2. Seleccionando voxels que serán semillas

Como ya dijimos, la materia gris está compuesta principalmente por neuronas, y la materia blanca por axones que las comunican. Como cada neurona posee asociado un axón, colocar semillas en la interfaz entre la materia gris y la blanca permite caracterizar las neuronas de la corteza [29] [16]. Cibu et al. [30] muestran que la materia blanca cercana a la materia gris está interconectada por pequeños axones. Como estamos interesados en realizar un estudio de las conexiones entre regiones distantes del cerebro, decidimos situar las semillas a $3mm$ de la corteza, evitando así el efecto de éstos axones locales. El problema es que la corteza del cerebro no es uniforme, sino que está llena de surcos y circunvoluciones. Calcular la distancia entonces no es inmediato, necesita un método que tome estas propiedades en cuenta. A continuación presentamos el método *Fast Marching Method* y como es posible utilizarlo para posicionar las semillas respetando la forma de la materia blanca.

Fast Marching Method es un método para resolver numéricamente una versión restringida de la ecuación *Eikonal*. La misma, en su forma general, es una ecuación diferencial no lineal

que se encuentra comúnmente en problemas de propagación de onda. Tiene la forma:

$$V(x)|\nabla u(x)| = F(x), x \in \Omega$$

Donde Ω es un subconjunto abierto de R^n con un *buen comportamiento* en su borde. $F(x)$ se denomina el costo temporal y $V(x)$ es la velocidad de la onda en cada punto. En el caso particular que queremos resolver $u(x_\omega) = 0, x \in \delta\Omega$; $F(x) = 1$ y $V(x) = 1$, por lo que la ecuación se resume a:

$$|\nabla u(x)| = 1, x \in \Omega$$

$u(v)$ en este caso representa el tiempo que tarda la onda en llegar desde algún elemento del borde hasta el punto v moviéndose a velocidad constante de una unidad de espacio por unidad de tiempo. Dada la forma de la velocidad, $u(v)$ también representa **la distancia mas corta que existe entre cualquier punto v de la imagen y el borde de Ω** . Dependiendo la orientación que se elija, las distancias a los puntos internos de la superficie serán negativas y las distancias a los puntos externos positivas (Figura 3.2). *FMM* resuelve este problema en tiempo $O(n \log(n))$ [31], siendo n la cantidad de voxels de la imagen.

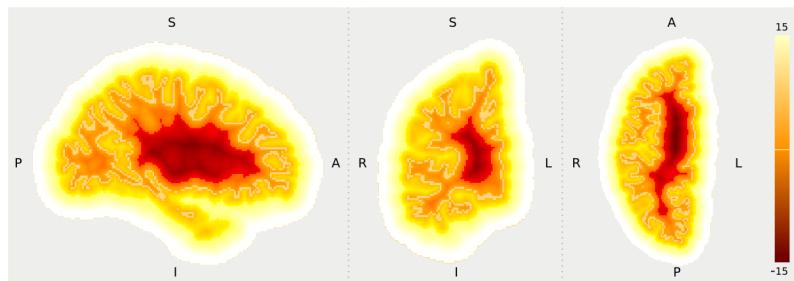


Fig. 3.2: *FMM* sobre el hemisferio derecho, el borde la materia blanca fue resaltado intensionalmente

Es posible utilizar este algoritmo para seleccionar voxels a cierta profundidad en la materia blanca. Usando como borde la corteza cerebral podemos crear un mapa de distancias en la materia blanca. El gradiente de este mapa de distancias es un campo vectorial donde cada vector apunta hacia el interior de la materia blanca. Caminar partiendo desde los puntos en la siguiendo este campo permite adentrarse respetando la morfología de la materia blanca. Una ventaja de este método es que permite guardar un mapeo entre cada coordenada de la superficie y la semilla que la representa. Otra ventaja es que es posible realizar todo el proceso en tiempo $O(n \log(n))$.

3.3. Creando tractogramas a partir de las semillas

Un tractograma es una imagen donde cada voxel representa la probabilidad de que ese punto del cerebro esté conectado a la semilla elegida mediante un conjunto de axones. Una forma de crear el tractograma de una semilla es generar un gran número de streamlines desde ella y luego calcular la frecuencia de visitas por cada voxel. Se denomina *streamline* al camino que puede realizar una partícula de agua siguiendo un mapa probabilístico de transiciones entre voxels. Es importante destacar que el estimar los tractogramas de esta

manera genera un sesgo respecto a la distancia. Cuanto más lejos está un voxel mayor es el número de transiciones probabilísticas necesarias para llegar a él. Esto provoca que los voxels lejanos tengan intrínsecamente valores pequeños.

Usando el método descrito en la sección anterior situamos semillas en la materia blanca de cada sujeto a 3mm de profundidad. Por cada semilla simulamos el recorrido de quince mil partículas de agua a través de la materia blanca. En base a los *streamlines* resultantes se crearon mapas de visitas para cada semilla. Un mapa de visitas es una imagen donde cada voxel posee el número de *streams* que lo conectan con la semilla. Luego usamos la transformación propuesta en Moreno-Dominguez et al. [15]:

$$T_i = \frac{\log(M_i + 1)}{\log(p + 1)}$$

Donde T_i es el valor del voxel i en el tractograma resultante; M_i es el valor del voxel i en el mapa de visitar y p es el número de partículas usadas. Esto nos permite evitar el sesgo producto de la distancia que obtendríamos al dividir por la cantidad de partículas utilizadas. Dado que $M_i \leq p, \forall i \in [1..N]$ cada voxel del tractograma tendrá un valor entre uno y cero. $T_i = 1$ representa que todos los *streams* pasaron por el voxel, mientras que $T_i = 0$ representa que ninguno pasó.

3.4. Clustering de Semillas: Estado del Arte.

Moreno-Dominguez et al. [15] implementan el algoritmo *Agglomerative Hierarchical Clustering* para agrupar los tractogramas. En este algoritmo, cada *feature* comienza en un cluster distinto. Luego, el algoritmo selecciona iterativamente dos clusters siguiendo algún criterio de similitud; los agrupa en un nuevo cluster y crea un elemento representativo de este. La jerarquía resultante de agrupar todos los clusters es expresada como un dendrograma. En el trabajo de Moreno-Dominguez utilizan como medida de similitud la distancia coseno (Ecuación 3.1) y como criterio de *linkage* el centroide (Ecuación 3.2).

$$\text{similarity}(X, Y) = 1 - \frac{X \cdot Y}{\|X\| \|Y\|} \quad (3.1) \quad \text{centroide}(X, Y) = \frac{n_X X + n_Y Y}{n_X + n_Y} \quad (3.2)$$

$$X, Y \in R^m, n_z = \#z$$

Para mejorar los resultados del *clustering* realizan distintos tipos de preprocesamiento en varias etapas. Aquí daremos solo una breve descripción de los mas relevantes, para mayores detalles favor de referirse al paper.

Una de las primeras modificaciones es al algoritmo *Agglomerative Hierarchical Clustering*. Dado un número k , las primeras k iteraciones sean entre clusters vecinos y de tamaño similar. Esto es, solo los clusters que se encuentran a menos de cierta distancia física en el cerebro pueden ser unidos. A su vez, solo se unen los clusters que poseen un tamaño similar para que el dendrograma crezca de manera balanceada. Una vez obtenido el dendrograma proceden a eliminar las inversiones dentro del mismo. Una inversión sucede cuando se unen dos clusters con una distancia interna mayor a la distancia entre ellos. Las inversiones no

cambian la jerarquía de los clusters, sino que solo complican la interpretación visual de los datos [32]. Una forma de eliminarlas es colapsando las ramas que la componen en una sola jerarquía con mas de dos elementos. Un ejemplo de inversión y el resultado de quitarla se muestran en las Figuras 3.3 y 3.4 respectivamente.

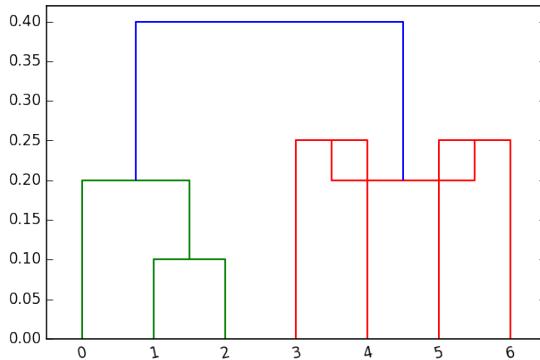


Fig. 3.3: Dendrograma de inversión.

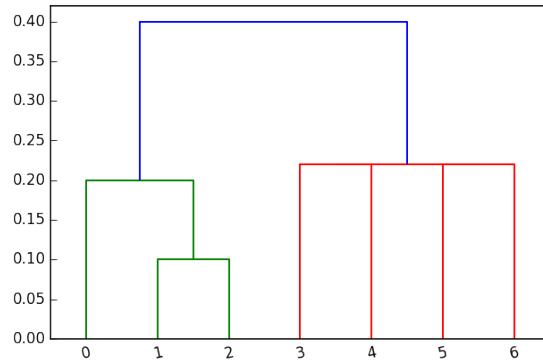


Fig. 3.4: Dendrograma con inversión corregida.

Otro paso de preprocesamiento es el quitar *outliers* del dendrograma. Esto en realidad lo hacen durante la etapa de *clustering*. Evitan que los clusters de un solo elemento se unan a otros clusters si la (di)similitud es mayor a cierto *threshold*. Al hacer este paso durante el *clustering* previenen que los *outliers* afecten la forma de los nuevos centroides.

Una vez finalizados todos los pasos el resultado es un dendrograma. Para parcelar la corteza solo es necesario seleccionar una altura en la cual cortar dicho dendrograma. Los clusters que estén por debajo de ese corte serán las distintas parcelas.

4. CONTRIBUCIONES

En el capítulo anterior presentamos cómo generar los tractogramas y el estado del arte para agruparlos. En este capítulo presentamos las contribuciones de nuestro trabajo. Comenzamos con una contribución menor que es estudiar la estabilidad de dos algoritmos de tractografía. Uno de ellos es el que luego fue utilizado en el resto del estudio, por lo que es importante determinar si es estable. Luego hacemos un análisis teórico del estado del arte y mostramos algunas de las problemáticas que posee. Finalmente utilizamos una función para transformar los tractogramas al espacio euclídeo y mostramos como esto permite mejorar tanto la complejidad espacial como temporal del algoritmo de *clustering*.

4.1. Estudio de la Convergencia de los Tractogramas

Algunas preguntas interesantes a realizar sobre los tractogramas son: ¿Al repetir el experimento, podremos obtener el mismo tractograma?; ¿Cuántas partículas son necesarias para ello? y ¿Qué tanto difieren los resultados entre los distintos algoritmos de tractografía?.

Para responder estas preguntas tomamos una implementación de tractografía ya existente llamada *LocalTracking* (LT de aquí en mas) y un algoritmo propio (MSL de aquí en más). Ambos algoritmos poseen una estructura similar: Encuadran la imagen de difusión en un modelo; en base a ese modelo crean un mapa de transiciones probabilísticas entre voxels y lo recorren de manera aleatoria hasta cumplir un criterio de parada. Para detalles sobre la implementación, por favor referirse al Anexo.

Para determinar si los algoritmos se estabilizaban y el número de partículas necesario para que eso suceda utilizamos la técnica estadística de *bootstrap* [34]. Bootstrap es una forma de aproximar la distribución del muestreo de un estadístico en base a calcular el mismo utilizando sucesivos remuestreos de los datos con repeticiones. Esto es especialmente útil cuando el número de muestras que se posee de la población no es significativamente alto. En nuestro caso situamos mas de setecientas semillas en el Área de Broca y luego generamos quince mil *streamlines* por cada una. Luego calculamos el tractograma medio y la varianza de cada voxel utilizando mil submuestras aleatorias del mismo tamaño. Esto se repitió con varios tamaños de submuestra para estudiar así la variabilidad a medida que la cantidad de partículas crecía.

4.2. Análisis del Método Moreno-Dominguez

Utilizar *Hierarchical Agglomerative Clustering* con la distancia coseno y el *linkage* centroide presenta al menos dos desventajas. Primero, la distancia coseno obliga a tener que comparar explícitamente cada centroide con los clusters existentes. Notemos que esto implica mantener los clusters en memoria. Por otro lado, el promedio de probabilidades no necesariamente representa una probabilidad [33]. Esto quiere decir que el centroide de

un grupo de tractogramas no necesariamente representa un tractograma.

4.2.1. Clustering de vectores colineales

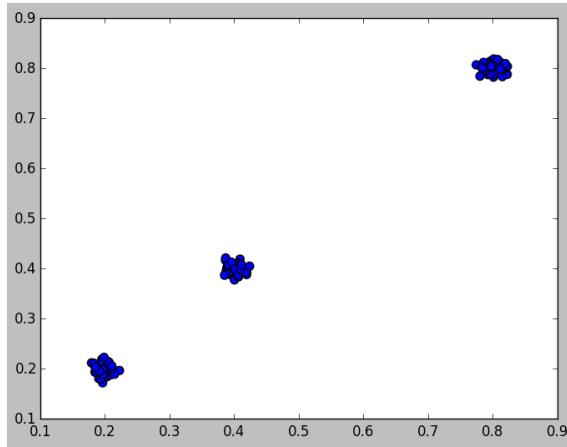


Fig. 4.1: Tres cluster colineales

La distancia coseno es una forma de medir correlación entre vectores. Por ello, cuando lo que se intenta agrupar son vectores colineales el resultado es aleatorio. Por ejemplo, cuando se aplica el procedimiento sobre los puntos de la Figura 4.1 el resultado del clustering es la Figura 4.2. Usando LogOdds y la métrica euclidiana se consigue el *clustering* de la Figura 4.6.

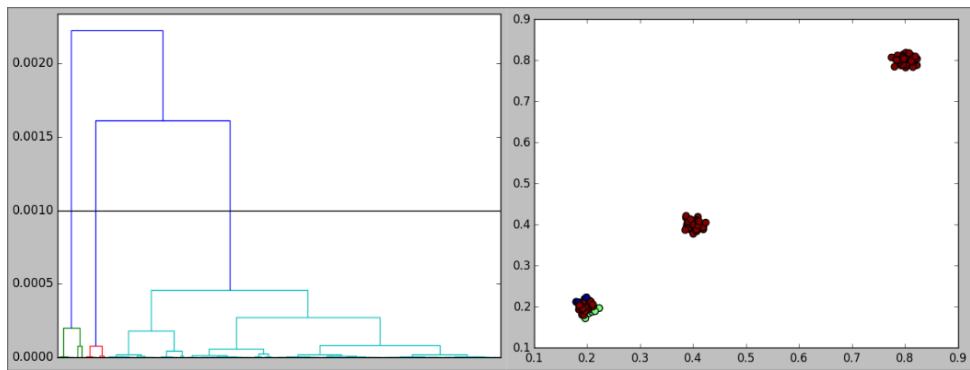


Fig. 4.2: Clustering resultado de utilizar el método Moreno

Los tractogramas son imágenes que por su naturaleza tienen muchos voxels con valores pequeños. La Figura 4.3 muestra el histograma de los valores en los tractogramas del Área de Broca. Esta gran cantidad de voxels con valores tan bajos podría generar ruido en las correlaciones.

4.2.2. Relación métrica-linkage

La Figura 4.4 muestra cuatro vectores, sus posiciones en coordenadas polares son: $p_1 = (0,4, 45^\circ)$; $p_2 = (0,3, 25^\circ)$; $p_3 = (0,4, 66^\circ)$ y $p_4 = (0,4, 4,5^\circ)$

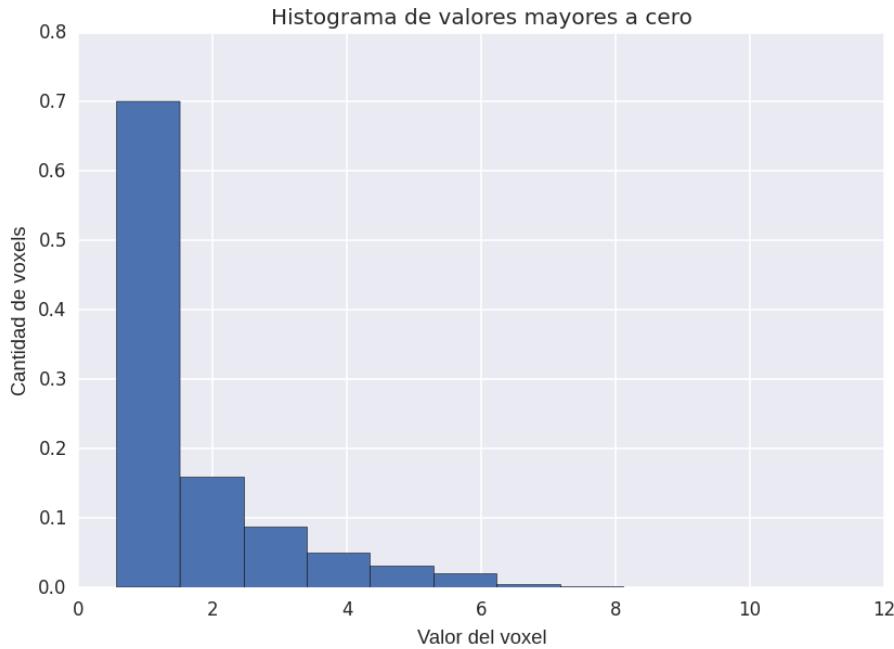


Fig. 4.3: Histograma normalizado de los valores en los tractogramas del Área de Broca

Podemos apreciar que al principio $d(p_2, p_3) < d(p_3, p_4) < d(p_1, p_2)$, siendo $d(x, y)$ la distancia coseno. Sin embargo, luego de utilizar el *linkage centroid* sucede que $d(p_1, p_c) < d(p_4, p_c)$. p_4 es ahora el punto que mas lejos está del centroide. Creando un representante p_m usando el ángulo medio entre p_2 y p_3 esto no sucede. Este fenómeno se da porque la distancia coseno tiene en cuenta el ángulo pero el centroide no. Por lo tanto el centroide no caracteriza al punto medio respecto a la distancia coseno.

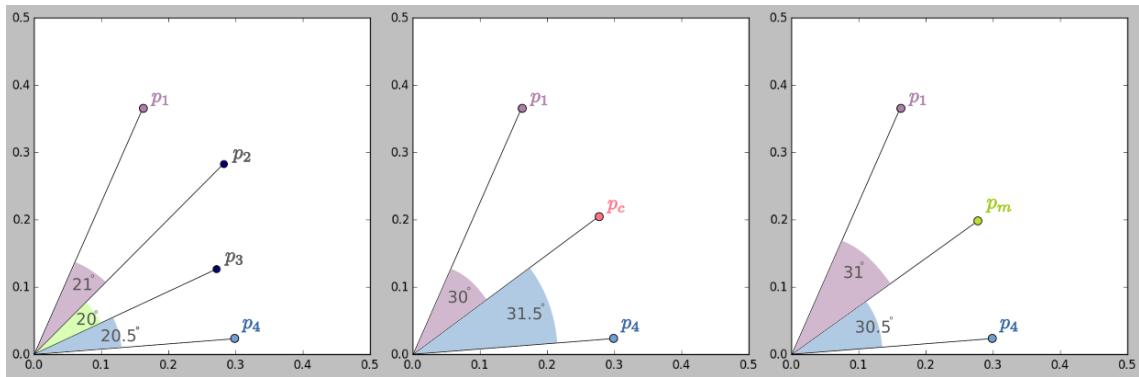


Fig. 4.4: El centroide no representa el punto medio respecto al ángulo

4.2.3. Complejidad algorítmica del Clustering

En todo el proceso el paso mas caro en términos computacionales es el *clustering*. En cada iteración del algoritmo es necesario comparar explícitamente cada nuevo centroide con todo el resto de los clusters. Esto es costoso computacionalmente. Por cada iteración es necesario hacer $O(c^2m)$ operaciones para recalcular todas las distancias, donde c es la cantidad de clusters y m es la longitud de los mismos. Dadas n semillas iniciales, la cantidad de iteraciones a realizar son $n - 1$. La complejidad de este método es $O(n^3m)$.

4.3. Clustering en el Espacio Euclídeo

En la sección anterior presentamos algunos de los inconvenientes teóricos del método propuesto por Moreno-Dominguez. A continuación mostramos como solucionar todos ellos haciendo uso de la transformación *logit*.

4.3.1. Transformación Logit

Sea P_M el espacio de una distribución discreta para M etiquetas:

$$P_M = \left\{ p \mid p = (p_1, \dots, p_n) \in (0, 1)^M, \sum p_i = 1 \right\}$$

La función $\text{logit}: P_M \rightarrow R^{M-1}$ define una transformación entre el espacio P_M y el espacio euclídeo R^{M-1} . Dados los vectores $Q \in P^M$ y $S \in R^{M-1}$:

$$S_i = \text{logit}(Q_i) = \log \left(\frac{Q_i}{Q_M} \right)$$

Para el caso de $M = 2$ permite transformar la distribución Bernoulli discreta al espacio euclídeo. Podemos ver en la ecuación 4.1 su expresión analítica y en la Figura 4.5 representación gráfica.

$$\text{logit}(p) = \log \left(\frac{p}{1-p} \right) \quad (4.1)$$

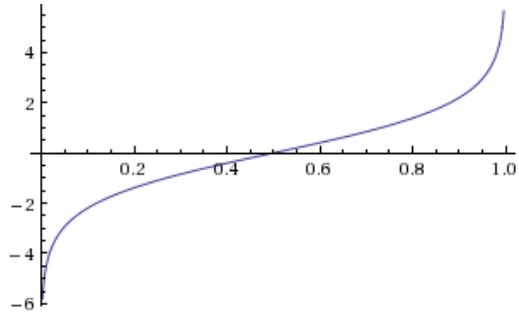


Fig. 4.5: Representación gráfica de la función logit

Trabajar en el espacio euclídeo nos asegura que la suma y la multiplicación por escalares están contenidos en el mismo espacio. Pohl et al. [33] utilizan esta propiedad para realizar operaciones lineales sobre mapas probabilísticos. A su vez, demuestran que la suma y multiplicación por escalares en el espacio euclídeo poseen un significado en el espacio de la distribución.

4.3.2. Modificando el Algoritmo de Clustering

Asumiendo que cada voxel v de un tractograma proviene de la variable aleatoria binaria:

$$X_v = \text{"La semilla está conectada con el voxel } v\text{"}$$

Es posible transformar el tractograma aplicando la función *logit* en cada uno de sus voxels. El resultado es un vector donde cada coordenada se encuentra en el espacio euclídeo. Más aún, las operaciones lineales entre los mismos voxels en distintos tractogramas están definidas. Esto nos permite usar la métrica euclídeana como función de similitud en *Agglomerative Hierarchical Clustering*.

Transformar de espacio los tractogramas y cambiar la función de similitud posee varias ventajas. Para empezar permite agrupar correctamente vectores colineales. La Figura 4.6 muestra el resultado de aplicar este método a los vectores de la Figura 4.1. Podemos apreciar que las tres poblaciones se encuentran correctamente separadas y bien definidas.

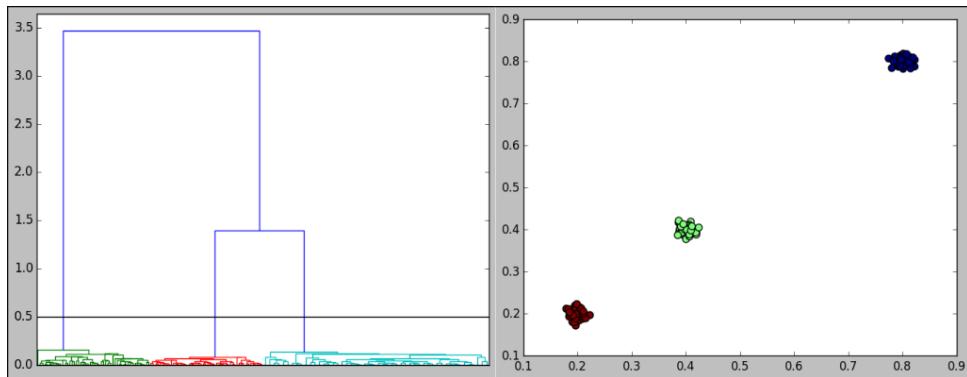


Fig. 4.6: Clustering resultado de utilizar el método *logit*

Otra ventaja es la buena relación métrica-*linkage*. Por definición el centroide es el centro de masa de los clusters. Esto quiere decir que es el punto que minimiza la distancia euclídea entre los clusters que lo componen. Por ende, el centroide caracteriza bien el punto medio de los vectores en el espacio euclídeo.

Finalmente, nuestro método también permite mejorar la complejidad algorítmica. Como ya explicamos, por cada iteración del algoritmo *Agglomerative Hierarchical Clustering* es necesario calcular un representante de la unión y luego computar su distancia al resto. Sin embargo, al usar la métrica euclídea junto con el *linkage* centroide es posible simplificar este paso. La fórmula de Lance y Williams permite computar las nuevas distancias sin comparar explícitamente los clusters. Esto baja significativamente la complejidad. Cada iteración pasa a costar $O(c^2)$ en vez de $O(c^2m)$, siendo c la cantidad de clusters y m la longitud de los mismos. Dadas n semillas iniciales, la complejidad temporal total del *clustering* es $O(n^3)$. Con la distancia coseno era $O(n^3m)$, siendo m la longitud de los tractogramas. Recordemos que en el contexto que estamos utilizando este algoritmo $m \gg n$. Por lo tanto, este resultado implica una gran mejora en la eficiencia del algoritmo.

4.3.3. Mejorando la Complejidad Espacial: Matrices Ralas

Durante la etapa de *clustering* es conveniente tener todos los tractogramas en memoria. En nuestro caso, la matriz con los **tractogramas sin transformar** del Área de Broca tiene dimensiones 762×3587328 . Asumiendo que cada valor se representa usando 8 Bytes, esta matriz ocupa un total de aproximadamente 20 Gigabytes. Sin embargo, solo un 1 % de los datos almacenados son no nulos. Esto implica que casi todo el espacio utilizado es desperdiciado.

Es posible mejorar esto eliminando de la matriz las columnas que poseen solo elementos nulos. La Figura 4.7 muestra la matriz que resulta de eliminar las columnas vacías de la matriz del Área de Broca. Si bien la nueva matriz posee dimensiones 762×121045 , aún solo el 27 % de los valores almacenados son no nulos. Manteniendo la representación de 8 Bytes es posible reducir el espacio necesario de 20 Gigabites a aproximadamente 700 Megabytes.

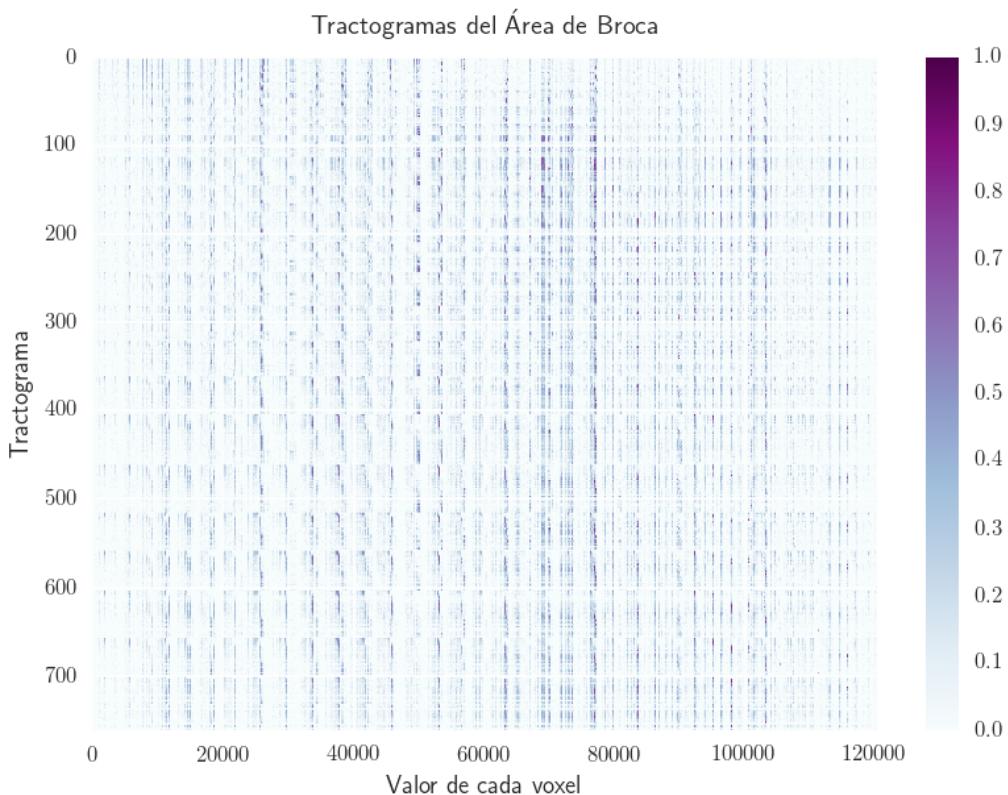


Fig. 4.7: Semillas en el hemisferio izquierdo.

El problema de este último método es que aún desperdicia mucho espacio. En el caso de utilizar todas los **tractogramas sin transformar** de un hemisferio, la matriz pasa a ser de dimensiones 21657×3587328 con un 1 % de valores no nulos. Para almacenar dicha matriz es necesario utilizar 587 Gigabytes. Por esto es necesario utilizar estructuras mas eficiente, que aprovechen lo ralo de las matrices. Ejemplos de estas estructuras son: *Dictionary of Keys*, o una matriz *Compressed Sparse Row* (CSR).

Si recordamos la forma que tiene la función logit encontramos un inconveniente al querer utilizar matrices ralas. Podemos ver en la Figura 4.5 que $\text{logit}(0) = -\infty$. Esto implica que la transformación de una matriz rala no es rala en términos de elementos nulos. Sin embargo podemos aprovechar ciertas propiedades para recuperar las matrices ralas. La distancia euclidena entre vectores es invariante a traslaciones lineales del sistema. Lo mismo sucede con las posiciones relativas de los centroides. Asignemos una representación finita c al valor $-\infty$. Un buen candidato para c es el $\log(\epsilon)$, donde ϵ es el *epsilon de la maquina*. Transformar todos los vectores y luego trasladarlos sumando c en cada componente dara como resultado una representación rala. Gracias a esto podemos utilizar DOK, CSR o cualquier estructura para reducir los costos espaciales del clustering.

5. RESULTADOS

5.1. Estabilidad Tractogramas

Las Figuras 5.1, 5.3 y 5.5 muestran, para tres semillas distintas, cinco cortes axiales del tractograma que se consigue al utilizar quince mil partículas.

Las Figuras 5.2, 5.4 y 5.6 muestran la varianza de cada voxel dentro de un mismo corte axial. La varianza se calculó generando mil tractogramas desde distinto número de streamlines.

La Figura 5.7 muestra la media y varianza de los voxels *A*, *B* y *C* marcados en las Figuras 5.2, 5.4 y 5.6. Estos voxels fueron los que mayor varianza presentaron al generar tractogramas con dos mil *streamlines*.

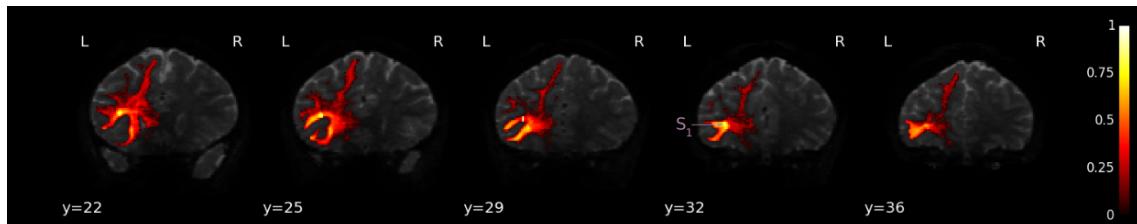


Fig. 5.1: Tractograma para la semilla *S1* utilizando toda la muestra.

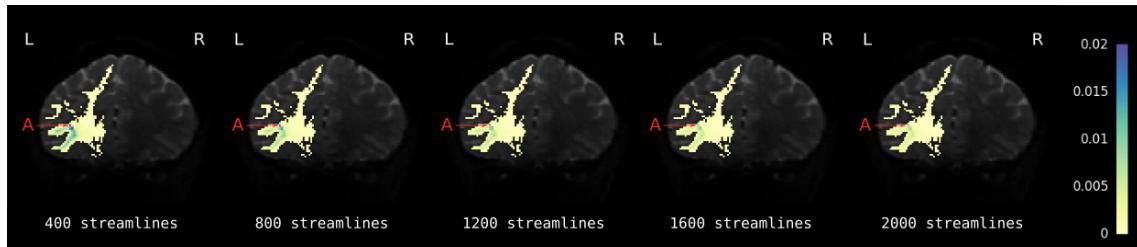


Fig. 5.2: Desviación Estandar respecto a la semilla *S1*. Mismo corte axial variando el tamaño de las submuestras.

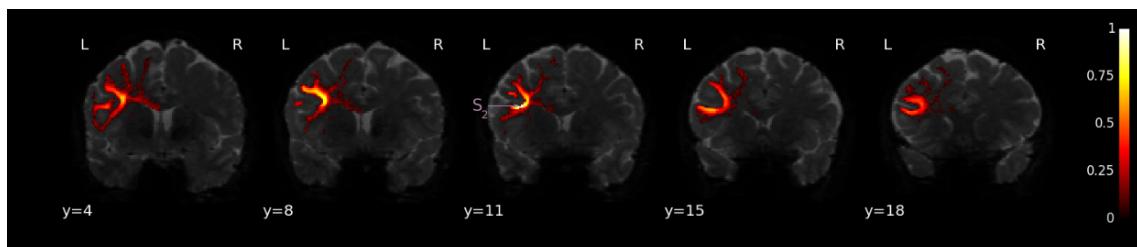


Fig. 5.3: Tractograma para la semilla *S2* utilizando toda la muestra.

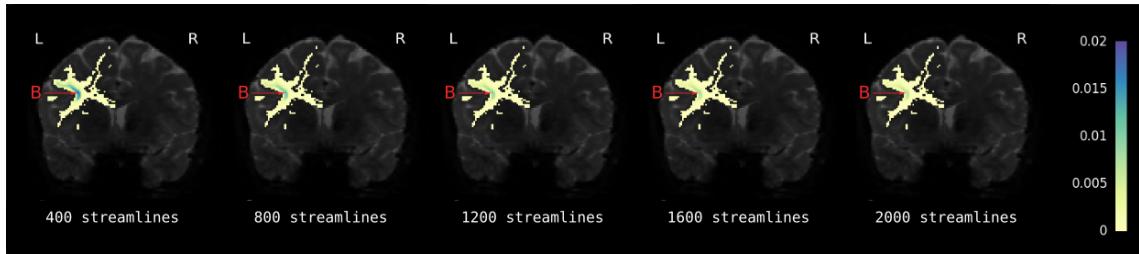


Fig. 5.4: Desviación Estandar respecto a la semilla S2. Mismo corte axial variando el tamaño de las submuestras.

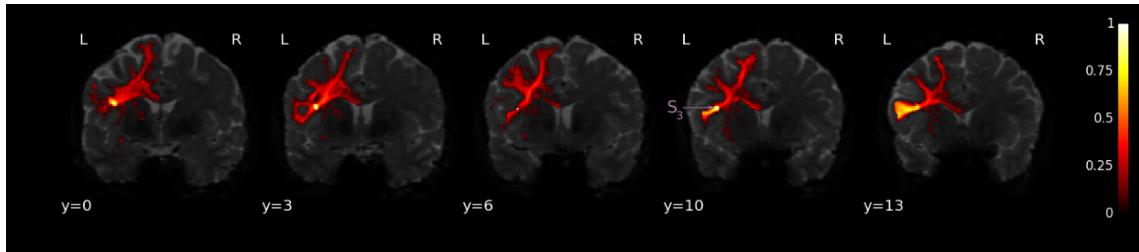


Fig. 5.5: Tractograma para la semilla S3 utilizando toda la muestra

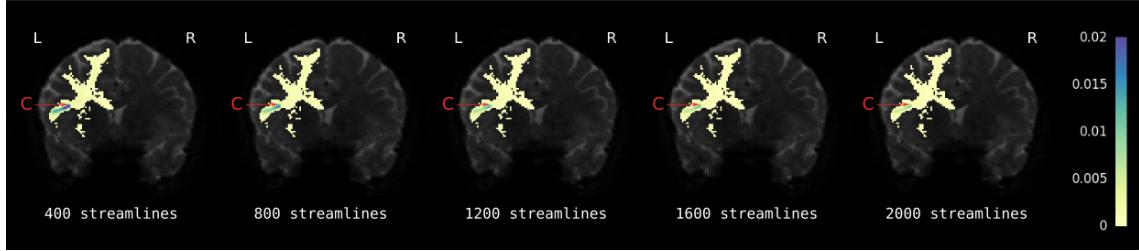


Fig. 5.6: Desviación Estandar respecto a la semilla S3. Mismo corte axial variando el tamaño de las submuestras.

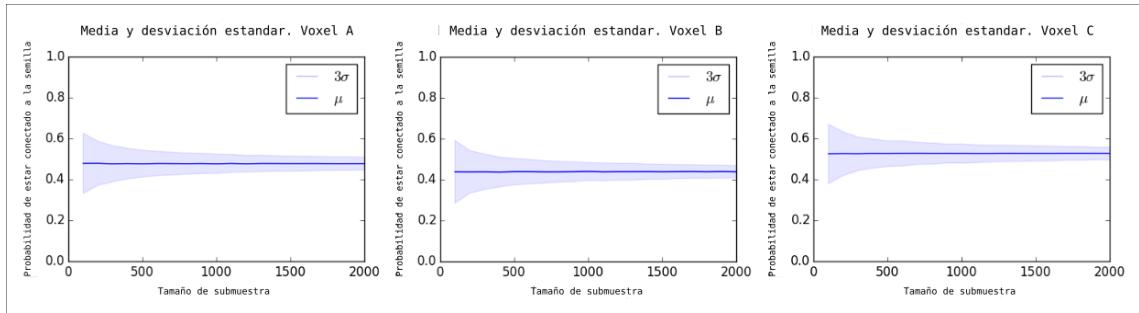


Fig. 5.7: Media y desviación estandar de los voxels con mayor varianza.

5.2. Posicionando semillas

Una forma rápida y sencilla de validar el correcto funcionamiento del algoritmo es, partiendo de una parcelación ya existente, pintar cada semilla resultante del color de la parcela de la cual se comenzó. Las semillas generadas deberían estar a la distancia deseada de la corteza, compartiendo el color de alguna de las parcelas más cercanas.

La Figura 5.8 muestra donde fueron posicionadas las semillas en la materia blanca del hemisferio izquierdo. El color de cada semilla representa el área del cual proviene.

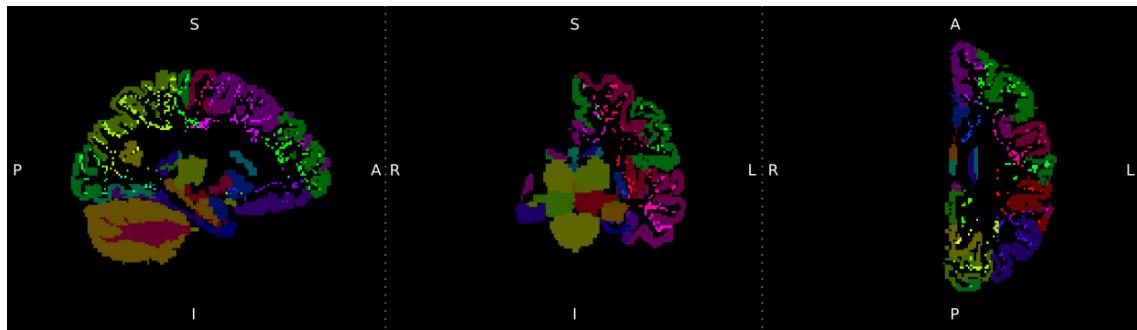


Fig. 5.8: Semillas en el hemisferio izquierdo.

5.3. Parcelando el Área de Broca

5.3.1. Distancia coseno con centroide

Las siguientes Figuras muestran los resultados obtenidos al parcelar el Área de Broca utilizando el método de Moreno-Dominguez.

5.3.2. Utilizando LogOdds

Las siguientes Figuras muestran los resultados obtenidos al parcelar el Área de Broca utilizando el método de Moreno-Dominguez. El threshold utilizado fue de 0,25. **Los resultados obtenidos luego de normalizar los vectores fueron tan malos que no vale la pena incluirlos.**

5.3.3. Lado a lado

5.4. Parcelando la Corteza

5.4.1. Utilizando LogOdds

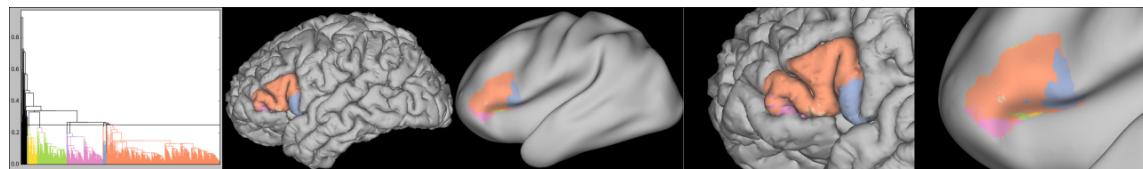


Fig. 5.9: Método Moreno sin preprocesamiento

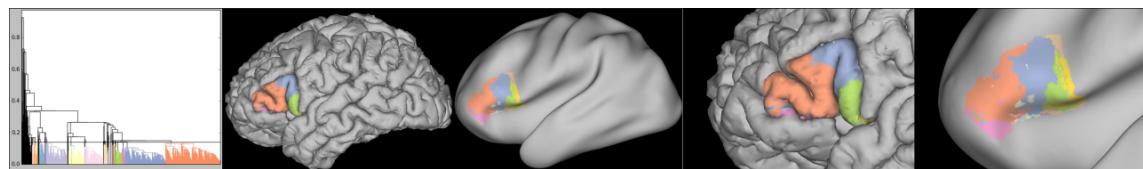


Fig. 5.10: Método Moreno sin preprocesamiento, mayor profundidad en el dendrograma

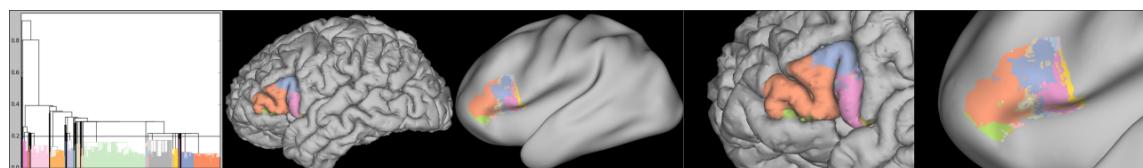


Fig. 5.11: Método Moreno, cuatrocientos pasos de preprocesamiento

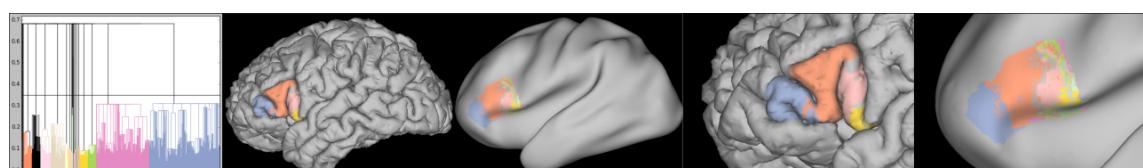


Fig. 5.12: Método Moreno, setecientos pasos de preprocesamiento

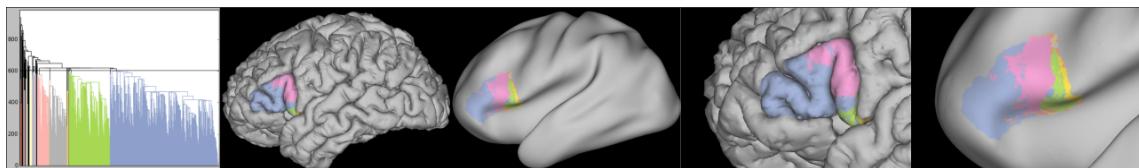


Fig. 5.13: Método Logit sin preprocessamiento

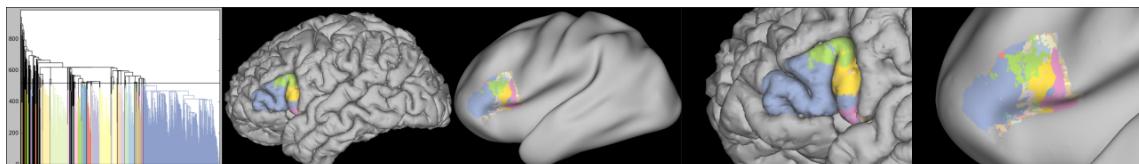


Fig. 5.14: Método Logit sin preprocessamiento, mayor profundidad en el dendrograma

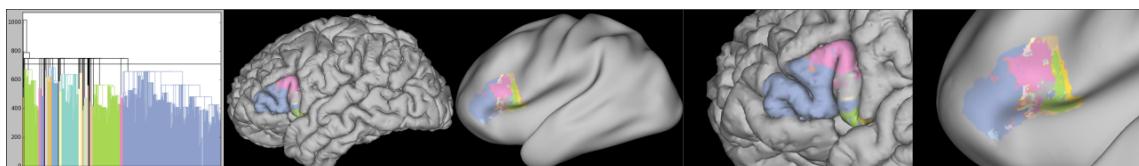


Fig. 5.15: Método Logit, cuatrocientos pasos de preprocessamiento

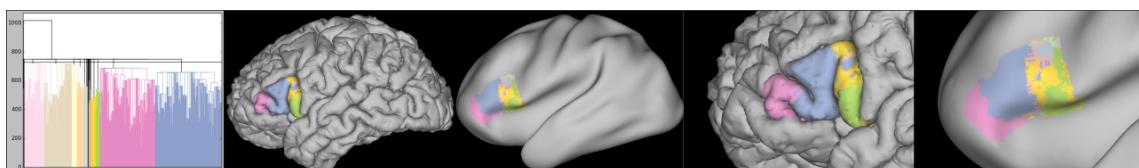


Fig. 5.16: Método Logit, setecientos pasos de preprocessamiento

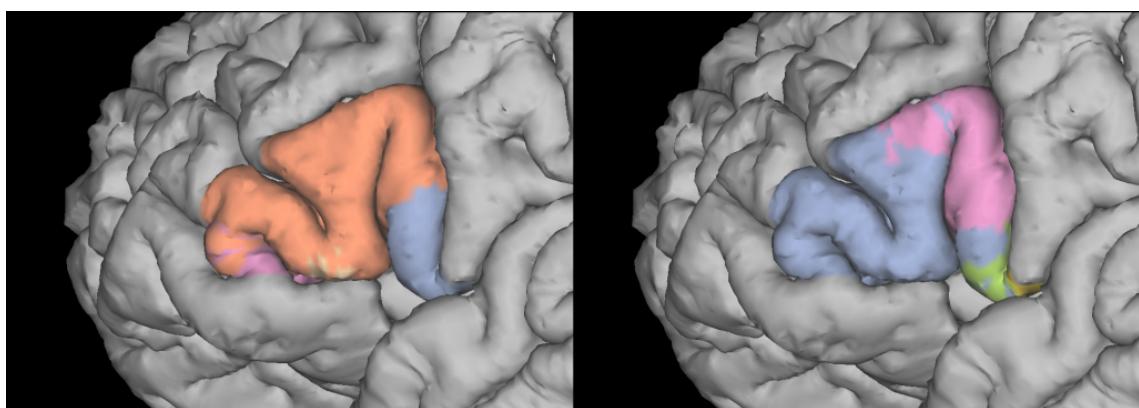


Fig. 5.17: Método Moreno (izquierda) y Logit (derecha) sin preprocessamiento

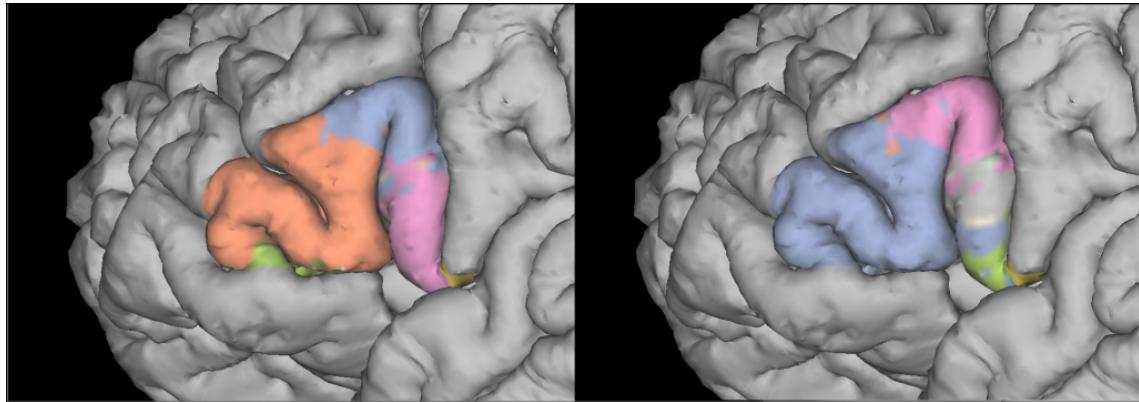


Fig. 5.18: Método Moreno (izquierda) y Logit (derecha). Cuatrocientos pasos de preprocesamiento

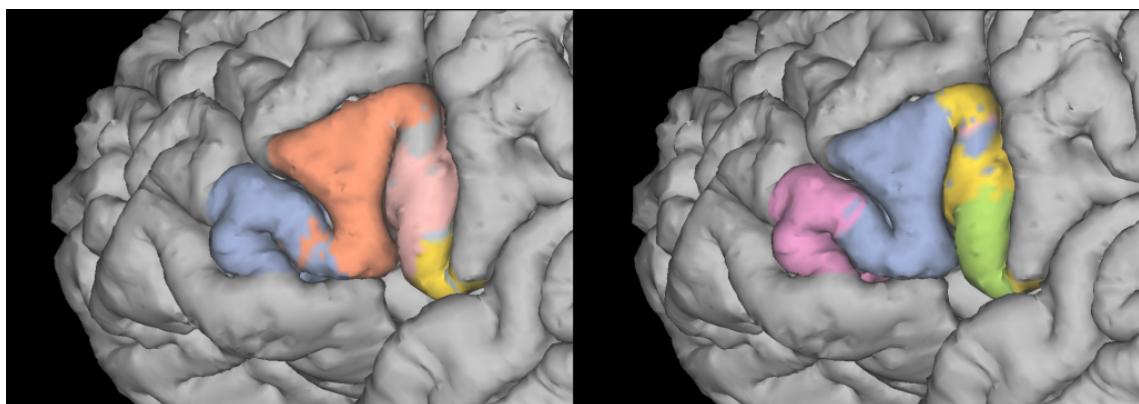


Fig. 5.19: Método Moreno (izquierda) y Logit (derecha). Setecientos pasos de preprocesamiento

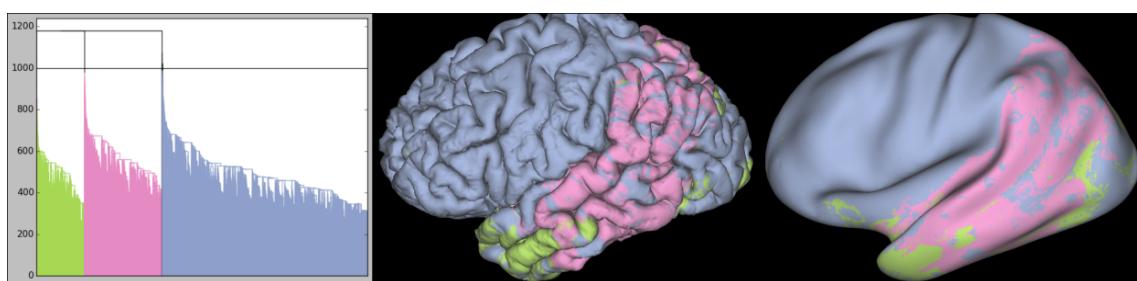


Fig. 5.20: Método Logit sin preprocessamiento

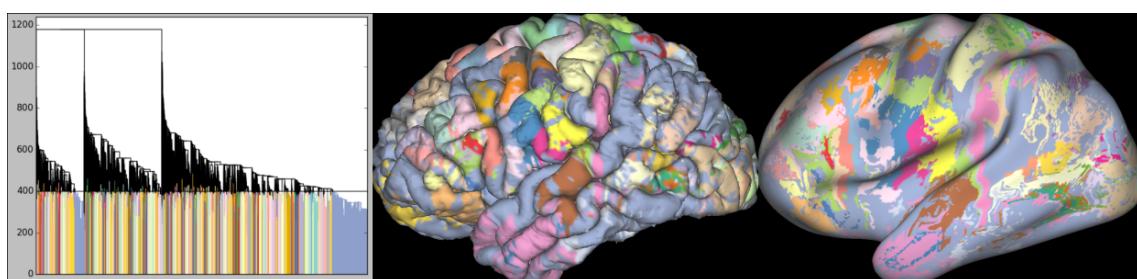


Fig. 5.21: Método Logit sin preprocessamiento, mayor profundidad en el dendrograma

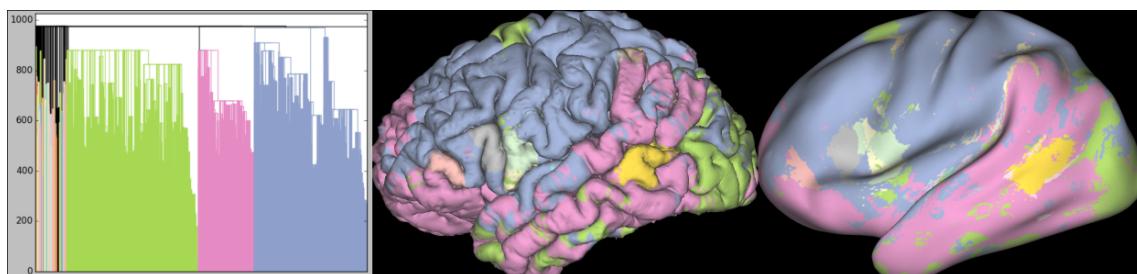


Fig. 5.22: Método Logit, veinte mil pasos de preprocessamiento

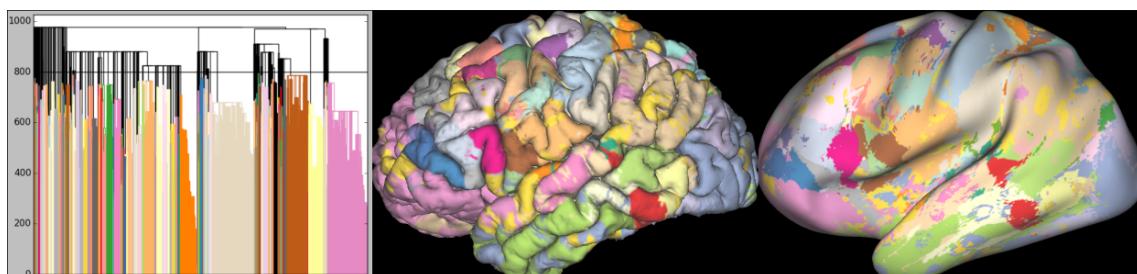


Fig. 5.23: Método Logit, veinte mil pasos de preprocessamiento, mayor profundidad

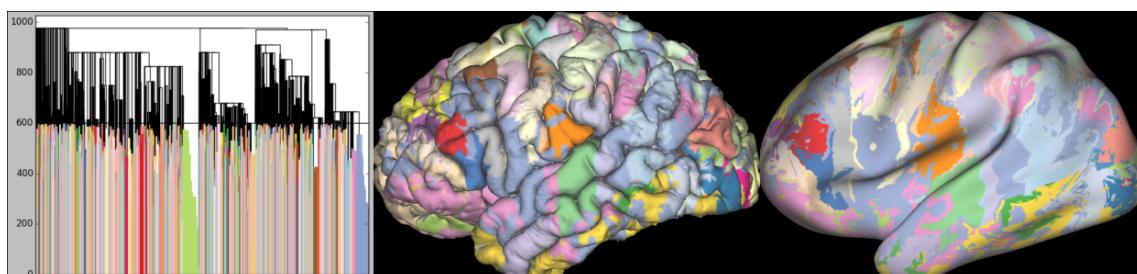


Fig. 5.24: Método Logit, veinte mil pasos de preprocessamiento, mayor profundidad

Bibliografía

- [1] K. Brodmann, “Vergleichende Lokalisationslehre der Großhirnrinde in ihren Prinzipien dargestellt aufGrund des Zellaufbaues,” 1909.
- [2] M. D. Greicius, B. Krasnow, A. L. Reiss, and V. Menon, “Functional connectivity in the resting brain: A network analysis of the default mode hypothesis,” *Proceedings of the National Academy of Sciences*, vol. 100, pp. 253–258, Jan. 2003.
- [3] D. M. Barch, G. C. Burgess, M. P. Harms, S. E. Petersen, B. L. Schlaggar, M. Corbetta, M. F. Glasser, S. Curtiss, S. Dixit, C. Feldt, D. Nolan, E. Bryant, T. Hartley, O. Footer, J. M. Bjork, R. Poldrack, S. Smith, H. Johansen-Berg, A. Z. Snyder, and D. C. Van Essen, “Function in the human connectome: task-fMRI and individual differences in behavior.,” *NeuroImage*, vol. 80, pp. 169–89, Oct. 2013.
- [4] S. M. Stufflebeam, H. Liu, J. Sepulcre, N. Tanaka, R. L. Buckner, and J. R. Madsen, “Localization of focal epileptic discharges using functional connectivity magnetic resonance imaging,” *Journal of Neurosurgery*, vol. 114, pp. 1693–1697, June 2011.
- [5] K. Oishi, A. V. Faria, P. C M van Zijl, and S. Mori, *MRI Atlas of Human White Matter*. Academic Press, Inc., 2010.
- [6] M. T. de Schotten, “Direct Evidence for a Parietal-Frontal Pathway Subserving Spatial Awareness in Humans,” *Science*, vol. 309, pp. 2226–2228, Sept. 2005.
- [7] J. Song, B. M. Young, Z. Nigogosyan, L. M. Walton, V. a. Nair, S. W. Grogan, M. E. Tyler, D. Farrar-Edwards, K. E. Caldera, J. a. Sattin, J. C. Williams, and V. Prabhakaran, “Characterizing relationships of DTI, fMRI, and motor recovery in stroke rehabilitation utilizing brain-computer interface technology,” *Frontiers in Neuroengineering*, vol. 7, p. 31, July 2014.
- [8] P. Dale, G. J. Augustine, D. Fitzpatrick, W. C. Hall, A.-S. LaMantia, J. O. McNamara, and L. E. White, *Neuroscience. 4th ed.* Sinauer Associates, 2008.
- [9] D. G. Taylor and M. C. Bushell, “The spatial mapping of translational diffusion coefficients by the NMR imaging technique,” *Physics in Medicine and Biology*, vol. 30, pp. 345–349, Apr. 1985.
- [10] P. Hagmann, L. Jonasson, P. Maeder, J.-p. Thiran, V. J. Wedeen, and R. Meuli, “Understanding Diffusion MR Imaging Techniques: From Scalar Diffusion-weighted Imaging to Diffusion Tensor Imaging and Beyond1,” *RadioGraphics*, vol. 26, pp. S205–S223, Oct. 2006.
- [11] M. Descoteaux, R. Deriche, T. Knosche, and A. Anwander, “Deterministic and Probabilistic Tractography Based on Complex Fibre Orientation Distributions,” *IEEE Transactions on Medical Imaging*, vol. 28, pp. 269–286, Feb. 2009.
- [12] S. Jbabdi, M. Woolrich, J. Andersson, and T. Behrens, “A Bayesian framework for global tractography,” *NeuroImage*, vol. 37, pp. 116–129, Aug. 2007.

- [13] J. a. Hartigan and M. a. Wong, "A K-Means Clustering Algorithm," *Journal of the Royal Statistical Society*, vol. 28, no. 1, pp. 100–108, 1979.
- [14] T. Hastie, R. Tibshirani, and J. Friedman, "Springer Series in Statistics The Elements of," *The Mathematical Intelligencer*, vol. 27, no. 2, pp. 83–85, 2009.
- [15] D. Moreno-Dominguez, A. Anwander, and T. R. Knösche, "A hierarchical method for whole-brain connectivity-based parcellation," *Human Brain Mapping*, vol. 35, pp. 5000–5025, Oct. 2014.
- [16] A. Anwander, M. Tittgemeyer, D. Von Cramon, A. Friederici, and T. Knosche, "Connectivity-Based Parcellation of Broca's Area," *Cerebral Cortex*, vol. 17, pp. 816–825, May 2006.
- [17] D. C. Van Essen, K. Ugurbil, E. Auerbach, D. Barch, T. E. J. Behrens, R. Bucholz, a. Chang, L. Chen, M. Corbetta, S. W. Curtiss, S. Della Penna, D. Feinberg, M. F. Glasser, N. Harel, a. C. Heath, L. Larson-Prior, D. Marcus, G. Michalareas, S. Moeller, R. Oostenveld, S. E. Petersen, F. Prior, B. L. Schlaggar, S. M. Smith, a. Z. Snyder, J. Xu, and E. Yacoub, "The Human Connectome Project: A data acquisition perspective," *NeuroImage*, vol. 62, no. 4, pp. 2222–2231, 2012.
- [18] P. J. Basser and E. Özarslan, "Introduction to Diffusion MR," in *Diffusion MRI* (H. Johansen-Berg and T. E. J. Behrens, eds.), ch. 1, pp. 2–10, Academic Press, Inc., 2009.
- [19] M. L. Lipton, "Introducing MRI," 2014.
- [20] F. Bloch, "Nuclear Induction," *Physical Review*, vol. 70, pp. 460–474, Oct. 1946.
- [21] H. C. Torrey, "Bloch Equations with Diffusion Terms," *Physical Review*, vol. 104, pp. 563–565, Nov. 1956.
- [22] E. O. Stejskal and J. E. Tanner, "Spin Diffusion Measurements: Spin Echoes in the Presence of a Time-Dependent Field Gradient," *The Journal of Chemical Physics*, vol. 42, no. 1, p. 288, 1965.
- [23] D. Le Bihan and E. Breton, "Imagerie de diffusion in vivo par résonance magnétique nucléaire," *Comptes rendus de l'Académie des sciences. Série 2, Mécanique, Physique, Chimie, Sciences de l'univers, Sciences de la Terre*, vol. 301, no. 15, pp. 1109–1112, 1985.
- [24] P. J. Basser, J. Mattiello, and D. Lebihan, "Estimation of the Effective Self-Diffusion Tensor from the NMR Spin Echo," *Journal of Magnetic Resonance, Series B*, vol. 103, pp. 247–254, Mar. 1994.
- [25] P. T. Callaghan, A. Coy, D. MacGowan, K. J. Packer, and F. O. Zelaya, "Diffraction-like effects in NMR diffusion studies of fluids in porous solids," *Nature*, vol. 351, pp. 467–469, June 1991.
- [26] D. S. Tuch, "Q-ball imaging," *Magnetic Resonance in Medicine*, vol. 52, pp. 1358–1372, Dec. 2004.

-
- [27] J.-D. Tournier, F. Calamante, D. G. Gadian, and A. Connelly, “Direct estimation of the fiber orientation density function from diffusion-weighted MRI data using spherical deconvolution,” *NeuroImage*, vol. 23, pp. 1176–1185, Nov. 2004.
 - [28] M. F. Glasser, S. N. Sotropoulos, J. A. Wilson, T. S. Coalson, B. Fischl, J. L. Andersson, J. Xu, S. Jbabdi, M. Webster, J. R. Polimeni, D. C. Van Essen, and M. Jenkinson, “The minimal preprocessing pipelines for the Human Connectome Project,” *NeuroImage*, vol. 80, pp. 105–124, 2013.
 - [29] S. Mori and P. C. M. van Zijl, “Fiber tracking: principles and strategies - a technical review,” *NMR in Biomedicine*, vol. 15, pp. 468–480, Nov. 2002.
 - [30] C. Thomas, F. Q. Ye, M. O. Irfanoglu, P. Modi, K. S. Saleem, D. a. Leopold, and C. Pierpaoli, “Anatomical accuracy of brain connections derived from diffusion MRI tractography is inherently limited,” *Proceedings of the National Academy of Sciences*, vol. 111, pp. 16574–16579, Nov. 2014.
 - [31] J. a. Sethian and A. Vladimirsky, “Ordered upwind methods for static Hamilton-Jacobi equations,” *Proceedings of the National Academy of Sciences*, vol. 98, pp. 11069–11074, Sept. 2001.
 - [32] F. Murtagh, *Multidimensional Clustering Algorithms*. Vienna: Comps Physica Verlag, 1985.
 - [33] K. M. Pohl, J. Fisher, S. Bouix, M. Shenton, R. W. McCarley, W. E. L. Grimson, R. Kikinis, and W. M. Wells, “Using the logarithm of odds to define a vector space on probabilistic atlases,” *Medical Image Analysis*, vol. 11, pp. 465–477, Oct. 2007.
 - [34] B. Efron, *The Jackknife, the Bootstrap and Other Resampling Plans*. Society for Industrial and Applied Mathematics, Jan. 1982.